Hara et al. BMC Bioinformatics 2010, 11:235
http://www.biomedcentral.com/1471-2105/11/235

BMC Bioinformatics

SOFTWARE Open Access

MTRAP: Pairwise sequence alignment algorithm by a new measure based on transition probability between two consecutive pairs of residues

Toshihide Hara*, Keiko Sato, Masanori Ohya

Abstract

Background: Sequence alignment is one of the most important techniques to analyze biological systems. It is also true that the alignment is not complete and we have to develop it to look for more accurate method. In particular, an alignment for homologous sequences with low sequence similarity is not in satisfactory level. Usual methods for aligning protein sequences in recent years use a measure empirically determined. As an example, a measure is usually defined by a combination of two quantities (1) and (2) below: (1) the sum of substitutions between two residue segments, (2) the sum of gap penalties in insertion/deletion region. Such a measure is determined on the assumption that there is no an intersite correlation on the sequences. In this paper, we improve the alignment by taking the correlation of consecutive residues.

Results: We introduced a new method of alignment, called MTRAP by introducing a metric defined on compound systems of two sequences. In the benchmark tests by PREFAB 4.0 and HOMSTRAD, our pairwise alignment method gives higher accuracy than other methods such as ClustalW2, TCoffee, MAFFT. Especially for the sequences with sequence identity less than 15%, our method improves the alignment accuracy significantly. Moreover, we also showed that our algorithm works well together with a consistency-based progressive multiple alignment by modifying the TCoffee to use our measure.

Conclusions: We indicated that our method leads to a significant increase in alignment accuracy compared with other methods. Our improvement is especially clear in low identity range of sequences. The source code is available at our web page, whose address is found in the section "Availability and requirements".

Background

Under a rapid increase of genome data, the need for accurate sequence alignment algorithms has become more and more important, and several methods have been developed. Sequence alignment algorithm is designed for mainly two purposes. One purpose is to design for comparing a query sequence with the database which contains preobtained sequences, and another is to design for generating multiple sequence alignment. FASTA [1] and BLAST [2], commonly used methods in molecular biology, are developed for database search, where a quick alignment algorithm is desired. For this quickness, the accuracy of alignment in these methods is lower than of the alignment by optimal algorithm.

In addition to for database search, sequence alignment is used for generating multiple alignment. In the multiple alignment, the accuracy is more important than the quickness. The recent popular multiple alignment methods, such as ClustalW [3], DIALIGN [4], TCoffee [5], MAFFT [6], MUSCLE [7], Probcons [8] and Probalign [9], are based on a "pairwise" alignment algorithm. In order to generate alignment with a realistic time and space costs, all of these methods use a progressive algorithm for constructing multiple alignment [10]. This "progressive" means to construct the multiple alignment by iterating pairwise alignment. These kind of methods give high accuracy for closely-related homologous sequences with identity more than 40%, but are not satisfied for distantly-related homologous sequences [11]. To improve the accuracy of the progressive algorithm, some measures based on, for instance, entropy

* Correspondence: hara@is.noda.tus.ac.jp
Department of Information Sciences, Tokyo University of Science, 2641 Yamazaki, Noda City, Chiba, Japan

[12] or consistency [13] have been developed. However, these measures are still not taken the intersite correlations of the sequences.

According to Anfinsen's dogma (also known as the thermodynamic hypothesis) [14], for a small globular protein, its three-dimensional structure is determined by the amino acid sequence of the protein. There may exist intersite correlations at least for two consecutive pairs of residues. Gonnet et al. considered this possibility [15]. We could improve alignment accuracy by taking into account information of the intersite correlations. Recently, Crooks et al. tried and tested such an approach [16], but they concluded that their approach is statistically indistinguishable from the standard algorithm. More recently, however, Lu and Sze proposed another approach [17], and they concluded that their strategy is able to consistently improve over existing algorithms on a few sets of benchmark alignments. Their approach is a kind of post processing algorithm. They take the average of the optimal values of the neighboring sites of one site, and they consider that the average value is the optimal value of that site. Note that they used usual "sum of pairs" measure for sequences. Their improvement of the accuracy in their tests was around 1~ 3% by using the BAliBASE 3.0 [18].

In this paper, we propose another approach introducing a new metric defined on compound systems of two sequences. Most of alignments are based on finding a path that gives the minimum value to the sum of difference (the maximum value to the sum of similarity) for each residue pair between two sequences. Our method is to change the way defining the difference (so, the sum) above by computing this sum of differences by introducing a quantity through the transition probability between consecutive pairs of residues. The comparison of our method with the method of Lu and Sze gives the following: In the very difficult range that the sequence identity is less than 15%, our method improves the accuracy nearly 8% up, but the Lu and Sze method improves it nearly 1% up.

A new measure taking the correlation of consecutive pairs of residues

First, let us establish some notations. Let Ω be the set of all amino acids, and Ω^* be the Ω with the indel (gap) "*": $\Omega^* \equiv \Omega \cup \{*\}$. Let $[\Omega^*]$ be the set of all sequences of the elements in Ω^*. We call an element of Ω a residue and an element of Ω^* a symbol. In addition, let $\Gamma \equiv \Omega \times \Omega$ be the direct product of two Ωs and $\Gamma^* \equiv \Omega^* \times \Omega^*$.

Consider two arranged sequences, $A = a_1 a_2 \ldots a_n$ and $B = b_1 b_2 \ldots b_n$, both of length n, where $a_i, b_j \in \Omega^*$. We also denote the sequences by $u_1 u_2 \ldots u_n$, where $u_i = (a_i, b_i) \in \Gamma^*$, and we call u_i a site in the following discussion. In general, the relative likelihood that the sequences are related as opposed to being unrelated is known as the "odds ratio":

$$R(A,B) = \frac{p(A;B)}{p(A)p(B)} = \frac{p(a_1,a_2,\cdots,a_n;b_1,b_2,\cdots,b_n)}{p(a_1,a_2,\cdots,a_n)p(b_1,b_2,\cdots,b_n)}. \quad (1)$$

Here, $p(a)$ is the occurrence probability of the given segment and $p(a; b)$ is the joint probability that the two segments occur. In order to arrive at an additive scoring system, Equation (1) is typically simplified by assuming that the substitutions are independent of the location and there is no intersite correlations; namely, $p(A) = \Pi\, p(a)$, $p(B) = \Pi\, p(b)$ and $p(A, B) = \Pi\, p(a, b)$. Thus the logarithm of Equation (1), known as the log-odds ratio, is now a sum of independent parts:

$$\log \frac{p(A;B)}{p(A)p(B)} = \sum_i s(a_i, b_i), \quad (2)$$

Where

$$s(a,b) = \log \frac{p(a;b)}{p(a)p(b)} \quad (3)$$

is the log likelihood ratio of the symbol pair (a, b) occurring as an aligned pair to that occurring as an unaligned pair. The $s(a, b)$ is called a score and $S = (s(a, b))$ is called a substitution matrix. These quantities (Equation (2) and (3)) are used to define a measure for pairwise sequence alignment [19]. Here, we define a normalized substitution matrix (i.e., every element in S takes the value between 0 and 1) and define a difference of A and B.

Let $f_s : [s_{\min}, s_{\max}] \mapsto R$ be a normalizing function:

$$f_s(x) \equiv \frac{s_{\max} - x}{s_{\max} - s_{\min}}, \quad 0 \le f_s(x) \le 1, \quad (4)$$

Where

$$s_{\max} \equiv \max\left\{ \max_{u \in \Gamma}\{S(u)\}, \text{gap cost} \right\}, \quad (5)$$

$$s_{\min} \equiv \min\left\{ \min_{u \in \Gamma}\{S(u)\}, \text{gap cost} \right\}. \quad (6)$$

Let put $\tilde{s}(a, b) = f(s(a, b))$ for $a, b \in \Omega$. This $\tilde{s}(a, b)$ is a normalized expression of the score $s(a, b)$. By using this quantity, we define a normalized substitution matrix as $M = (\tilde{s}(a, b))$. Then a difference of A and B is defined by

$$d_{\text{sub}}(A,B) = \sum_i \tilde{s}(a_i, b_i). \quad (7)$$

When the sequence A is equal to B the difference $d_{sub}(A, B)$ has a minimum value 0.

One of the essential assumption for the above approach (using a sum of independent parts as a difference of A and B) was the induction of the occurring probability. We could take more informative approach by including the intersite correlations. Crooks et al. tried one of such an approach [16]. They introduced a measure for two sequences based on a multivariate probability approximated by using the intersite relative likelihood. But, they concluded that their approach is statistically indistinguishable from the standard algorithm. We feel that their measure (equation (4) in their paper) is different from ours. To introduce their measure, they defined a type of joint probability. However it can not be a probability, bacause their quantity is the multiplication of likelihood "ratios", so it goes beyond more than 1. Moreover, we think that the intersite relative likelihood may not describe the difference of sequence A and B. Under an assumption that each site of the sequences has Markov property, we propose a new measure for two sequences by adding a transition effect and its weight ε (a degree of mixture):

$$R_{\text{our}}(A,B) = R(A,B)^{1-\varepsilon} R_t(A,B)^{\varepsilon}, \tag{8}$$

where

$$R_t(A,B) \equiv \prod_{i=1}^{n-1} p(u_{i+1} \setminus u_i). \tag{9}$$

Here we introduce a normalized transition $\tilde{t}(u_i, u_{i+1})$ called "Transition Quantity", in order to simplify the equation. Let $\tilde{t}(u_i, u_{i+1})$ be a normalized transition defined as

$$\tilde{t}(u_i, u_{i+1}) \equiv f_t(t(u_i, u_{i+1}); u_i), \tag{10}$$

$$t(u_i, u_{i+1}) \equiv \log p(u_{i+1} \setminus u_i), \tag{11}$$

where $f_t(x; u)$ is a normalizing function:

$$f_t(x;u) = \begin{cases} \dfrac{-x}{\max_{v\in\Gamma^*}\{-t(u,v)\}}, & \text{if } x > 0 \\ 1, & \text{otherwise} \end{cases} \tag{12}$$

By using the above quantity, a difference of A and B representing the "intersite transition" is defined as

$$d_{\text{trans}}(A,B) = \sum_{i=1}^{n-1} \tilde{t}(u_i, u_{i+1}). \tag{13}$$

Consequently, we define a difference measure for two sequences by combination of two differences d_{sub} and d_{trans}:

$$d_{\text{MTRAP}}(A,B) = (1-\varepsilon) d_{\text{sub}}(A,B) + \varepsilon d_{\text{trans}}(A,B). \tag{14}$$

Estimation of the Transition Quantity

Let us discuss how to estimate the transition quantity $\tilde{t}$. We can estimate the transition quantity by collecting reliable aligned protein sequences. In this study, we estimated the transition quantity by means of the superfamilies subset of the dataset SABmark (version 1.63) [20]. This set covers the entire known fold space using only high-quality structures taken from the SCOP database [21]. For a large set, the same sequences are re-used in the set. In order to reduce the bias introduced by multiple use of the same sequences, we assign a weight to each sequence. This approach is similar to the one described in the paper [22]. If a sequence occurs N times in the dataset, its weight is $N^{-1/2}$. We estimated the transition quantity from the weighted frequencies of observed transitions as follows.

Let $\mathcal{A}^{L,N}$ be the set of N sequences with length L:

$$\mathcal{A}^{L,N} = \{A_i \mid A_i = a_{i1}\cdots a_{iL} \in [\Omega^*], i = 1, \cdots, N\}.$$

Let $w_{ij}^k = (a_{ik}, a_{jk})$ be in the finite set Γ^* and it is a symbol pair in the kth site of the $\mathcal{A}^{L,N}$. In addition, let $\mathcal{A}$ be the set of all given sets $\mathcal{A}^{L,N}$ (i.e., the superfamilies set of the SABmark), and let N_A be the frequency of the sequence A in the set $\mathcal{A}$.

Let $C_{\mathcal{A}^{L,N}} : \Gamma^* \times \Gamma^* \mapsto \mathbf{R}$ be a mapping which represents the weighted frequency appearing the symbols (u, v) in $\mathcal{A}^{L,N}$ such that

$$C_{\mathcal{A}^{L,N}}(u,v) = \sum_{k=1}^{L-1}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \delta_{u,w_{ij}^k}\delta_{v,w_{ij}^{k+1}} N_{A_i}^{-1/2} N_{A_j}^{-1/2}, (u,v\in\Gamma^*), \tag{15}$$

$$\delta_{p,q} = \begin{cases} 1, & p = q \\ 0, & p \neq q \end{cases} \tag{16}$$

Let $p(v \setminus u)$ be a transition probability from the symbol pair u to the pair v on $\mathcal{A}$ such that

$$p(v \setminus u) = \begin{cases} \dfrac{\sum_{x\in\mathcal{A}} C_x(u,v)}{\sum_{x\in\mathcal{A}}\sum_{z\in\Gamma^*} C_x(u,z)}, & \text{if } \sum_{x\in\mathcal{A}}\sum_{z\in\Gamma^*} C_x(u,z) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

for u, v in the finite set Γ^*.

We define a matrix T = ($\tilde{t}$ (*u*,*v*)) called "Transition matrix" by the elements $\tilde{t}$ (*u*, *v*) as

$$\tilde{t}(u,v) = f_t(t(u,v);u),$$

where *t* (*u*, *v*) = log *p* (*v**u*) and f_t is the normalizing function defined by the equation (12).

MTRAP Algorithm

The MTRAP (sequence alignment method by a new Measure based on TRAnsition Probability) is an alignment algorithm by minimizing the value of a certain objective function based on the transition quantity (Figure 1). We describe the algorithm by means of dynamic programming [23].

Let *A*, *B* be two amino acid sequences such as

$$A : a_1 a_2 \cdots a_m,$$
$$B : b_1 b_2 \cdots b_n,$$

where a_i, $b_j \in \Omega$. Take the lattice point $P_k = (i_k, j_k)$, $i = 1, \ldots, m$, $j = 1, \ldots, n$ as in Figure 2. We call the sequence of the lattice points

$$\mathcal{R} = \{P_1, P_2, \cdots, P_N\} \tag{18}$$

a "route" with an initial point $P_1 = (0, 0)$ and a final point $P_N = (m, n)$ if the following conditions are met:

$$i_{k-1} \le i_k, j_{k-1} \le j_k, P_{k-1} \ne P_k \text{ for any } k(= 2, 3, \cdots, N). \tag{19}$$

Let α_R, β_R be maps from a route $\mathcal{R} = \{P_1, P_2, \ldots, P_N\}$ to a set Ω^* such that

$$\alpha_R(P_k) = \begin{cases} a_{ik} & i_k \ne i_{k-1} \quad (k = 2, \cdots, N) \\ * \quad k = 1 \quad \text{or} \quad i_k = i_{k-1} & (k = 2, \cdots, N) \end{cases} \tag{20}$$

$$\beta_R(P_k) = \begin{cases} b_{jk} & j_k \ne j_{k-1} \quad (k = 2, \cdots, N) \\ * \quad k = 1 \quad \text{or} \quad j_k = j_{k-1} & (k = 2, \cdots, N) \end{cases} \tag{21}$$

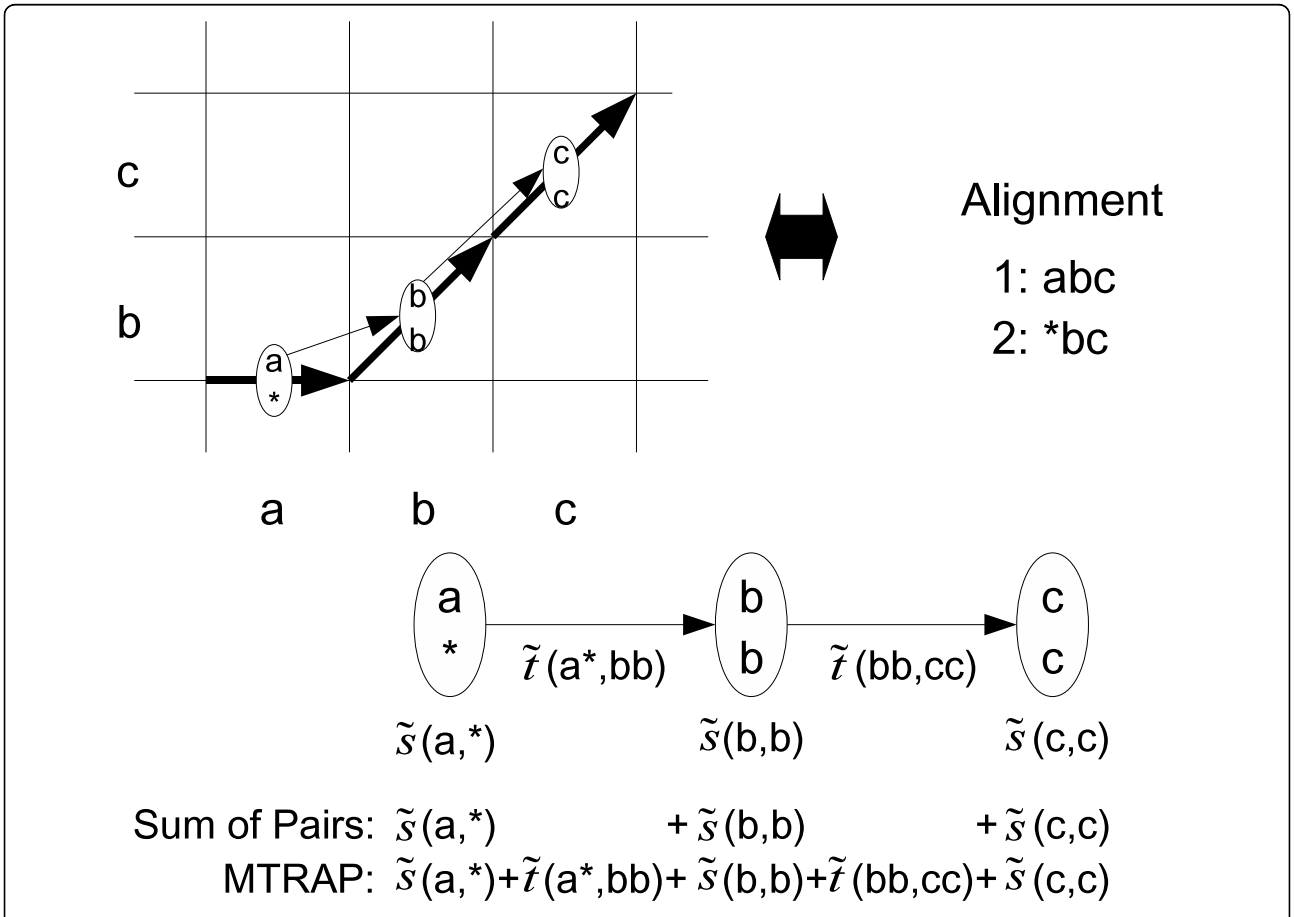

Figure 1 Illustration of MTRAP. The set of bold arrows indicates a possible route for alignment and the normal arrows indicate the transitions from a pair to the subsequent pair. A value $\tilde{s}$ (*u*) indicates a difference of two residues in a given pair *u* and $\tilde{t}$ (u_1, u_2) indicates the quantity based on a transition probability from a pair u_1 to the subsequent pair u_2. A conventional objective function such as "Sum of Pairs" is defined only by the sum of the value $\tilde{s}$ (*u*) of all possible pairs across the sequences, whereas our objective function in the MTRAP is defined by the sum of both $\tilde{s}$ (*u*) and $\tilde{t}$ (u_1, u_2) across the sequences (Equation (14)).

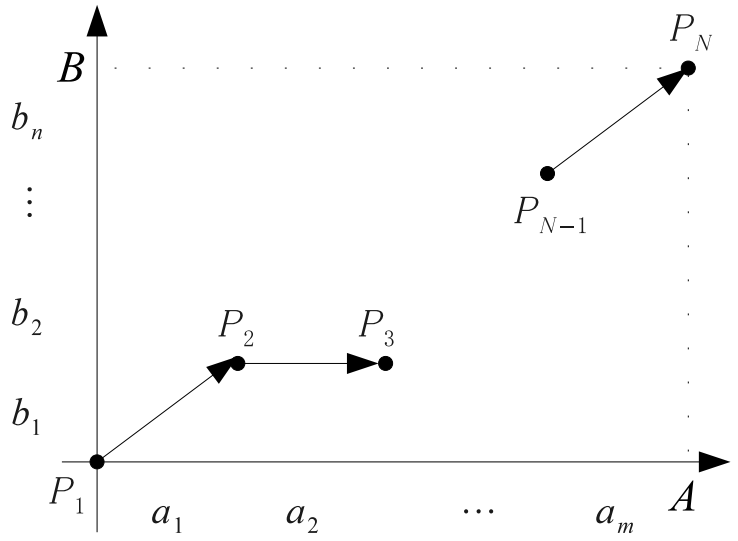

Figure 2 Lattice points with two-sequences. The input amino acid sequences $A = (a_1 \ldots a_m)$ and $B = (b_1 \ldots b_n)$ are placed on each two axes. An initial point P_1 (0,0) and a final point P_N (m, n) are fixed.

and μ_R be a map from the route $\mathscr{R}$ to the set of all symbol pairs $\Gamma^*(\equiv \Omega^* \times \Omega^*)$ such that

$$\mu_R(P_k) = (\alpha_R(P_k), \beta_R(P_k)). \tag{22}$$

We call the following A^* and B^* the alignment of A and B by the route $\mathscr{R}$:

$$\begin{array}{lcccccc} A^* & : & \alpha_R(P_1) & \alpha_R(P_2) & \cdots & \alpha_R(P_N), \\ B^* & : & \beta_R(P_1) & \beta_R(P_2) & \cdots & \beta_R(P_N). \end{array}$$

Let $R(P)$ be the set of all routes with the final point P, that is,

$$R(P) = \{\{P_1, \cdots, P_k\}; P_k = P\}. \tag{23}$$

Let us fix the following notations for the following discussion: (1) $\Gamma^{*-} \equiv \Omega^* \times \Omega$, (2) $\Gamma^{-*} \equiv \Omega \times \Omega^*$, (3) $\Gamma^{g-} \equiv \{*\} \times \Omega$, (4) $\Gamma^{-g} \equiv \Omega \times \{*\}$, (5) w_{open} is a constant called gap "opening" cost; $0 \le w_{\text{open}} \le 1$, (6) w_{extend} is a constant called gap "extending" cost; $0 \le w_{\text{extend}} \le w_{\text{open}}$ and (7) ε is a weight, $0 \le \varepsilon \le 1$ (i.e., the mixture of usual difference d_{sub} and our new difference d_{trans}). The difference between A and B by a route $\mathscr{R}$ is given by

$$d(\mathcal{R}) = \sum_{k=2}^{N} \tilde{d}_s(P_{k-1}, P_k; \mathcal{R}), \tag{24}$$

where d_s is a function from $\Gamma^* \times \Gamma^*$ to $\mathbf{R}$ such that

$$\tilde{d}_s(P_{k-1}, P_k; \mathcal{R}) = \begin{cases} d_e(\mu_R(P_{k-1}), \mu_R(P_k)), & k \ge 3 \\ d_i(\mu_R(P_k)), & k = 2 \end{cases} \tag{25}$$

$$\tilde{d}_i(u) = \begin{cases} (1-\varepsilon)\tilde{s}(u), & u \in \Gamma \\ (1-\varepsilon)w_{\text{open}}, & u \notin \Gamma \end{cases} \tag{26}$$

$$\tilde{d}_e(u_1, u_2) = \begin{cases} (1-\varepsilon)\tilde{s}(u_2) + \varepsilon\tilde{t}(u_1, u_2), u_1 \in \Gamma^*, u_2 \in \Gamma \\ (1-\varepsilon)w_{\text{open}} + \varepsilon\tilde{t}(u_1, u_2), \begin{cases} u_1 \in \Gamma^{-*}, u_2 \in \Gamma^{g-} \\ u_1 \in \Gamma^{*-}, u_2 \in \Gamma^{-g} \end{cases} \\ (1-\varepsilon)w_{\text{extend}} + \varepsilon\tilde{t}(u_1, u_2), \begin{cases} u_1 \in \Gamma^{g-}, u_2 \in \Gamma^{g-} \\ u_1 \in \Gamma^{-g}, u_2 \in \Gamma^{-g} \end{cases} \end{cases} \tag{27}$$

The degree of difference between A and B with respect to a final point P can be defined as

$$D(P) = \min_{R \in R(P)} \{d(\mathcal{R})\}. \tag{28}$$

Hence the degree of difference between A and B is

$$D_{AB} = D(P = (m, n)). \tag{29}$$

We calculate D_{AB} by a dynamic programming technique as below. For a final point $P_k = (i, j)$ and a route $\mathscr{R} = \{P_1, ..., P_k\} \in R(P_k)$, we have

$$P_k = (i, j) \text{ and } P_{k-1} = Q_1 \text{ or } Q_2 \text{ or } Q_3, \tag{30}$$

where $Q_1 = (i - 1, j)$, $Q_2 = (i - 1, j - 1)$ and $Q_3 = (i, j - 1)$. Therefore

$$R(P_k) = R_1(P_k) \cup R_2(P_k) \cup R_3(P_k) \tag{31}$$

with

$$R_l(P_k) = \{\{P_1, \cdots, Q_l, P_k\}; \{P_1, \cdots, Q_1\} \in R(Q_1)\} \tag{32}$$

for $l = 1, 2, 3$. Thus we obtain

$$D(P) = (m, n) = \min_{R \in R(P)} \{d(R)\} \tag{33}$$

$$= \min_{l=1,2,3} \min_{R \in R_l(P)} \{d(R)\} \tag{34}$$

$$= \min_{l=1,2,3} \min_{\substack{R_2 \in R(Q_l) \\ R_1 \in R_l(P)}} \{d(R_2) + d_s(Q_l, P; R_1)\} \tag{35}$$

$$= \min\{D_1(P), D_2(P), D_3(P)\}, \tag{36}$$

where

$$D_l(P_k = (i, j)) \equiv \min_{\substack{R_2 \in R(Q_l) \\ R_1 \in R_l(P_k)}} \{d(R_2) + d_s(Q_l, P; R_1)\} \tag{37}$$

for $l = 1, 2, 3$.

Each point Q_l has three points Q_l^1, Q_l^2, Q_l^3 which possibly go to Q_l one step after. These points are precisely written as

$$\begin{bmatrix} Q_1^1 & Q_1^2 & Q_1^3 \\ Q_2^1 & Q_2^2 & Q_2^3 \\ Q_3^1 & Q_3^2 & Q_3^3 \end{bmatrix} = \begin{bmatrix} (i-2, j) & (i-2, j-1) & (i-1, j-1) \\ (i-2, j-1) & (i-2, j-2) & (i-1, j-2) \\ (i-1, j-1) & (i-1, j-2) & (i, j-1) \end{bmatrix} \tag{38}$$

when $Q_1 = (i - 1, j)$, $Q_2 = (i - 1, j - 1)$, $Q_3 = (i, j - 1)$.

The distances D_l ($P_k = (i, j)$) can be obtained from one step before by the following recursion relations:

$$D_l(P_k = (i, j)) = \min_{R_1 \in R_l(P_k)} \min_{p=1,2,3} \min_{\substack{R_3 \in R(Q_l^p) \\ R_2 \in R_p(Q_l)}} \{d(R_3) + d_s(Q_l^p, Q_l; R_2) + d_s(Q_l, P_k; R_1)\} \tag{39}$$

$$= \min_{R_1 \in R_l(P_k)} \min_{p=1,2,3} \{D_p(Q_l) + d_s(Q_l, P_k; R_1)\} \tag{40}$$

$$= \min_{\substack{p=1,2,3 \\ R_1 \in R_l(P_k)}} \{D_p(Q_l) + d_s(Q_l, P_k; R_1)\} \tag{41}$$

for $l = 1, 2, 3$. The values D_l of initial point and those of the edge points are assumed as

$$D_2((0, 0)) = 0, \tag{42}$$

$$D_l((0, 0)) = \infty \text{ for } l = 1, 3, \tag{43}$$

$$D_l((1, j)) = \infty \text{ for } l = 1, 2, j = 1, \cdots, n, \tag{44}$$

$$D_l((i, 1)) = \infty \text{ for } l = 2, 3, i = 1, \cdots, m, \tag{45}$$

Moreover for other special cases, the recursive relation of the edge points satisfies

$$D_1(P_k = (i, 1)) = D_1(P_{k-1}) + d_s(P_{k-1}, P_k; \mathcal{R}) \tag{46}$$

for $\mathscr{R} \in R_1(P_k)$, $i = 1, ..., m$,

$$D_3(P_k = (1, j)) = D_3(P_{k-1}) + d_s(P_{k-1}, P_k; \mathcal{R}) \tag{47}$$

for $\mathscr{R} \in R_3(P_k)$, $j = 1, ..., n$.

This calculation is completed in mn steps.

Multiple sequence alignment by MTRAP

In order to discuss the effect of using MTRAP algorithm in the iteration step of progressive multiple alignment, we modified the TCoffee [5], a consistency-based progressive multiple alignment program, by means of our distance (Equation (14)). TCoffee constructs a primary library (pairwise alignments between all of the sequences to be aligned) at first step. We implemented our algorithm to make this primary library. That is, our modified TCoffee constructs a multiple alignment by following steps.

1. Generating a primary library by using MTRAP
2. Extending the library (Calculate a consistency)
3. Making a guide tree for the progressive step
4. Constructing a multiple alignment by progressive strategy

The modified TCoffee uses the extended library obtained by the MTRAP algorithm for aligning.

Performance evaluation

We compared the performance of MTRAP to those of the most often used nine methods: Needle, ClustalW2,

MAFFT, TCoffee, DIALIGN, MUSCLE, Probcons, Probalign and TCoffee-Lu/Sze. The details of these nine methods are: (1) Needle, a global pairwise alignment using Needleman-Wunsch algorithm [24] contained in EMBOSS package ver. 5.0.0 [25]; (2) ClustalW2 [3,26], a typical progressive multiple alignment method; (3) MAFFT ver. 6 [6], a fast method with Fourier transform algorithm; (4) TCoffee ver. 5.31 [5], a heuristic consistency-based method that combines global and local alignments; (5)

DIALIGN ver. 2.2 [4], a method with segment-segment approach; (6) MUSCLE ver. 3.7 [7], a method with Log-Expectation algorithm; (7) Probcons ver. 1.12 [8], a probabilistic consistency-based method, (8) Probalign ver. 1.1 [9], a multiple sequence alignment using partition function posterior probabilities and (9) TCoffee-Lu/Sze, an improved TCoffee modified by the Lu/Sze algorithm [17]. These programs without MAFFT used their default parameters and MAFFT used "L-IN-i" strategy mode.

To measure the accuracy of each method, we used three different databases: HOMSTRAD (version November 1, 2008) [27,28], PREFAB 4.0 [7] and BAliBASE 3.0 [18]. These are the databases of structure-based alignments for homologous families. We used the all 630 pairwise alignments obtained from the HOMSTRAD for pairwise alignment tests, and used the all 1031 multiple alignments obtained from this database for multiple alignment tests. We also used the all 1682 protein pairs obtained from the PREFAB 4.0 for pairwise alignment tests. The BAliBASE 3.0 contains 5 different reference sets of alignment for testing multiple sequence alignment methods. We used the BBS sets included in the references 1,2,3 and 5. The BBS sets are described as being trimmed to homologous regions.

In order to avoid using the same dataset for training and test, We estimated the transition quantity by using the superfamilies subset from the dataset SABmark, which is described in the section "Estimation of the Transition Quantity". We also used this dataset for optimizing the parameters w_{open}, w_{extend}, ϵ. Consequently, MTRAP uses the followings for parameter values: $w_{\text{open}} = f_s$ (-11), $w_{\text{extend}} = f_s$(-0.3) and $\varepsilon = 0.775$.

Alignment accuracy was calculated with the Q (quality) score [7]. The Q score is defined as the ratio of the number of correctly aligned residue pairs in the test alignment (i.e., the alignment obtained by a specified algorithm such as MTRAP, Needle, etc.) to the total number of aligned residue pairs in the reference alignment. When all pairs are correctly aligned, the score have a maximum value 1, and when no-pairs are aligned the score have a minimum value 0. This score has previously been called the SPS (Sum of Pairs Score) [29] or the developer score [30]. Let us redefined this score in our notations. Let A_i (i = 1, i..., N) indicates the ith sequence of the reference alignment with length L, and let $a_{ik} \in \Omega^*$ indicates the kth symbol in the sequence A_i. When $a_{ik} \neq *$, it is important to find the number of the site in the test sequence corresponding to the symbol a_{ik}, whose numbers are denoted by n_{ik}. When $a_{ik} = *$, put $n_{ik} = 0$ (i = 1, ..., N). Then the Q score is given as

$$\text{Q score} = \frac{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{k=1}^{L}\Delta_{a_{ik},a_{jk}}\delta_{n_{ik},n_{jk}}}{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{k=1}^{L}\Delta_{a_{ik},a_{jk}}},$$

$$\Delta_{x,y} = \begin{cases} 1, & x \neq * \quad \text{and} \quad y \neq * \\ 0, & x = * \quad \text{or} \quad y = * \end{cases}.$$

Implementation

The MTRAP algorithm is implemented as a C++ program. The program has been tested in several types of Linux machines including 32-bit ×86 platform and also has been tested on Mac OSX snow leopard (64-bit). The program has a number of command-line options, e.g., the option setting the value of a parameter such that $\tilde{s}$, $\tilde{t}$, ε, w_{open}, w_{extend}, and the option controlling the output format. The program accepts only multiple-fasta format as an input format.

Results and Discussion

Performance evaluation of pairwise alignment

We compared MTRAP with nine different alignment methods including the modified TCoffee by using all 1682 protein pairs of PREFAB 4.0 and all 630 protein pairs of HOMSTRAD. We used GONNET250 matrix with the MTRAP. The similarity between the test alignment (sequence alignment by each method) and the reference alignment (obtained from PREFAB 4.0 or HOMSTRAD) was measured with the Q score. Suppose as usual that the reference alignment is the optimal alignment, the results of PREFAB 4.0 (Table 1) and those of HOMSTRAD (Table 2) indicate that our method works well compared with other methods. Our method achieves the highest ranking compared with all other methods except only one range 30-45%. Especially for the identity range 0-15%, MTRAP is 4 ~ 5% accurate than the 2nd ranking method. For the identity range 30-45%, Probcons and Probalign perform slightly better (~ 1%).

Performance evaluations using other substitution matrices

We did the performance evaluations using three different substitution matrix series: PAM, BLOSUM and GONNET, with HOMSTRAD and PREFAB 4.0, whose

Table 1 Average Q scores on the PREFAB 4.0 database.

Method	PREFAB 4.0				
	0-15% (212)	15-30% (458)	30-45% (74)	All (1682)	CPU
MTRAP[a]	**0.248**	**0.674**	0.877	**0.615**	120
MAFFT	0.170	0.671	0.860	0.568	200
DIALIGN[b]	0.133	0.556	0.814	0.518	100
MUSCLE	0.205	0.632	0.867	0.581	35
ClustalW2	0.199	0.644	0.859	0.586	70
Probcons	0.204	0.647	0.875	0.590	120
Probalign	0.195	0.654	**0.887**	0.593	100
TCoffee	0.198	0.642	0.872	0.585	180
TCoffee-Lu/Sze	0.198	0.647	0.874	0.588	270

The average Q scores of four testing datasets with different identity ranges on PREFAB 4.0 are shown. The number in parentheses denotes the number of alignments in each sequence identity range. For each sequence identity range, the best scores are in bold. CPU is the total computing time for all alignments in seconds.

[a]MTRAP uses GONNET250 substitution matrix.

[b]DIALIGN reported critical errors for some testing data. Therefore, the scores of DIALIGN were calculated by the partial testing data.

Table 2 Average Q scores on the HOMSTRAD database (Pairwise only).

Method	HOMSTRAD (Pairwise only)				
	0-15% (25)	15-30% (207)	30-45% (173)	All (630)	CPU
MTRAP[a]	**0.412**	**0.659**	0.879	**0.819**	45
MAFFT	0.309	0.610	0.863	0.796	60
DIALIGN[b]	0.216	0.546	0.825	0.760	35
MUSCLE	0.337	0.625	0.868	0.802	15
ClustalW2	0.313	0.619	0.867	0.800	25
Probcons	0.344	0.650	0.884	0.816	50
Probalign	0.325	0.649	**0.886**	0.818	40
TCoffee	0.341	0.634	0.872	0.809	70
TCoffee-Lu/Sze	0.347	0.649	0.879	0.815	100

The average Q scores of four testing datasets with different identity ranges on HOMSTRAD are shown. The notations are the same as Table 1.

results are shown in Table 3 and Figure 3, respectively. We compared MTRAP with two typical global alignment programs, Needle and ClustalW2, which can use various substitution matrices. We used all 630 protein pairs of HOMSTRAD and all 1682 protein pairs of PREFAB 4.0. The similarity between the test alignment and the reference alignment was measured with the Q score.

For every typical substitution matrix (i.e., PAM250, BLOSUM62 and GONNET250), MTRAP has more than 80% accuracy (e.g., 0.817 with PAM250 and BLOSUM62), whereas Needle and ClustalW2 have less than 80% accuracy (e.g., Needle has 0.763 with PAM250 and 0.768 with BLOSUM62) (Table 3). Moreover, it is important to notice that for two sequences with less than 30% sequence identity, our method improves the alignment accuracy significantly. For instance, MTRAP

Table 3 Average Q scores in pairwise alignment tests with typical substitution matrices.

Matrix / Method	HOMSTRAD (Pairwise only)			
	0-15% (25)	15-30% (207)	30-45% (173)	All (630)
PAM250				
MTRAP	**0.421**	**0.655**	**0.874**	**0.817**
Needle	0.226	0.548	0.837	0.763
ClustalW2	0.234	0.528	0.817	0.747
BLOSUM62				
MTRAP	**0.410**	**0.653**	**0.878**	**0.817**
Needle	0.223	0.556	0.843	0.768
ClustalW2	0.276	0.585	0.861	0.784
GONNET250*				
MTRAP	**0.412**	**0.659**	**0.879**	**0.819**
ClustalW2	0.313	0.619	0.867	0.800

The average Q scores of four testing datasets with different identity ranges on HOMSTRAD are shown. The number in parentheses denotes the number of alignments in each sequence identity range. For each sequence identity range, the best scores in each substitution matrix are in bold.

*Needle does not support GONNET matrix.

with PAM250 matrix has 0.421 for 0-15% sequence identity and 0.655 for 15-30% sequence identity, and ClustalW2 with PAM250 matrix has 0.234 for 0-15% sequence identity and 0.528 for 15-30% sequence identity, respectively.

Figure 3 shows the results with another database PREFAB 4.0 that are the ratios of the average Q scores for each identity range. For all substitution matrices, these three programs show almost the same alignment accuracy when the sequence identity is more than 60%, whereas the ratio clearly shows that MTRAP has higher accuracy than other programs in decreasing the sequence identity within 0-60%. For instance, MTRAP and Needle with PAM120 have 0.356 and 0.152 for 0-20% sequence identity, and those with BLOSUM80 have 0.363 and 0.166, respectively. For alignments with sequence identity 0-20%, the average Q score of MTRAP is 1.5-2.3 times more accurate than that of Needle. Moreover, MTRAP outperforms ClustalW2 at the same range by 1.4, 1.3 and 1.1-1.2 times for PAM, BLOSUM and GONNET series, respectively.

Performance of MTRAP algorithm for multiple alignment

We modified the TCoffee by means of our MTRAP algorithm. Table 4 and Table 5 show the accuracy of the modified TCoffee (TCoffee-MTRAP) compared with other methods including the original TCoffee with HOMSTRAD and BAliBASE 3.0. For all testing datasets, TCoffee-MTRAP shows the performance increase over the original TCoffee. Especially for the identity range 0-15%, TCoffee-MTRAP is 8.0% more accurate than the original TCoffee with HOMSTRAD, whereas the TCoffee modified by the Lu/Sze algorithm (TCoffee-Lu/Sze) is

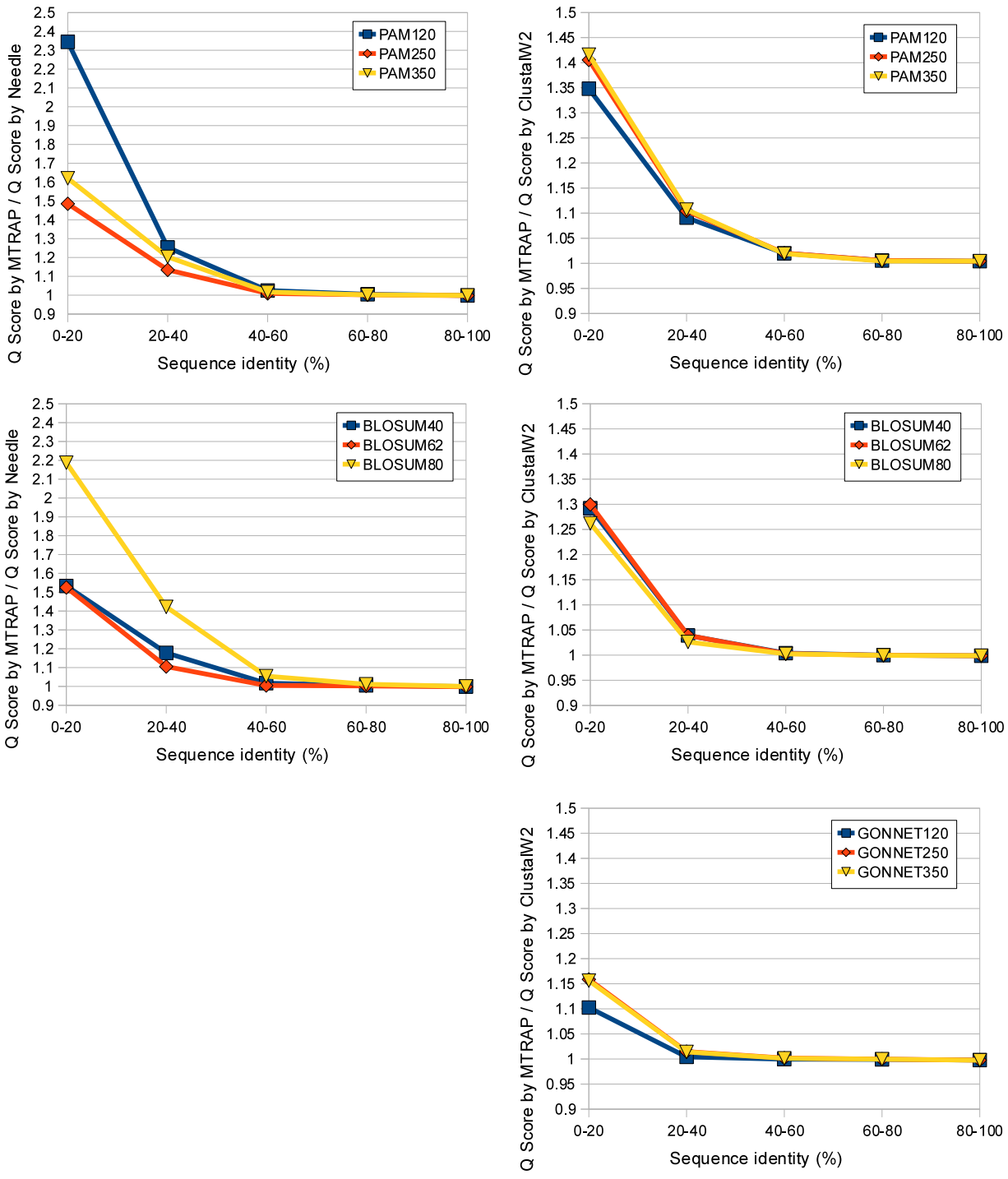

Figure 3 The ratios of the average Q scores on the PREFAB 4.0 database. The upper two figures show the ratio of the average Q score by MTRAP to that by Needle and the ratio of ours to that by ClustalW2, both for PAM substitution matrix. The middle two figures show the ratios for BLOSUM substitution matrix. The last figure shows the ratio for GONNET substitution matrix.

Table 4 Average Q scores on the HOMSTRAD database.

Method	HOMSTRAD			
	0-15%(32)	15-30%(325)	30-45%(331)	All(1031)
TCoffee-MTRAP	**0.395**	**0.666**	**0.868**	**0.819**
TCoffee-Lu/Sze	0.322	0.648	**0.868**	0.813
TCoffee	0.315	0.642	0.864	0.809
MAFFT	0.288	0.632	0.858	0.803
DIALIGN*	0.203	0.559	0.811	0.761
MUSCLE	**0.333**	0.643	0.860	0.809
ClustalW2	0.313	0.628	0.855	0.815
Probcons	0.329	0.666	0.873	0.820
Probalign	0.310	**0.670**	**0.877**	**0.824**

The average Q scores of four testing datasets with different identity ranges on HOMSTRAD are shown. For each sequence identity range, the better scores of the TCoffee modified by Lu/Sze algorithm (TCoffee-Lu/Sze) and the TCoffee modified by MTRAP algorithm (TCoffee-MTRAP) are in bold. The best scores of the other methods are also in bold.

*DIALIGN reported critical errors for some testing data. Therefore, the scores of DIALIGN were calculated by the partial testing data.

0.7% more accurate than the original (Table 4). Also for V1 (i.e., the sequence identity is less than 20%) of the reference 1, TCoffee-MTRAP is 6.0% more accurate than the original TCoffee on BAliBASE 3.0, whereas TCoffee-Lu/Sze is 0.3% more accurate than the original (Table 5). In some domains, the two methods Probcons and Probalign, both of which are based on the probabilistic consistency strategy, are more accurate than TCoffee-MTRAP. Note that these two methods use the parameter values estimated from the BAliBASE 2.0 database.

Conclusions

MTRAP is a global alignment method that is based on a new metric. The metric is determined by an adjusted substitution matrix and a transition probability-based matrix between two consecutive pairs of residues including gap-residue derived from structure-based alignments.

We indicated here that our approach, which takes into account an intersite correlation on the sequences, leads to a significant increase in the alignment accuracy, especially, for the low identity range. We also indicated that the MTRAP improves the alignment accuracy for any substitution matrices. Moreover, we confirmed that our algorithm works well together with a consistency based progressive approach for constructing multiple alignment.

However, the methods Probcons and Probalign were more accurate than TCoffee-MTRAP in some multiple alignment tests. The probabilistic consistency strategy is an improved consistency strategy of TCoffee. Therefore, combining MTRAP pairwise algorithm with the probabilistic consistency strategy will generate more high quality multiple alignments. We will examine this fact in a separate paper.

MTRAP has the similar calculation cost with other pairwise methods. That is, MTRAP has $O(mn)$ calculation order for two input sequences with length m and n. Our CPU time shown in the Tables 1, 2 are almost the same as others.

Pairwise sequence alignment is among the most important technique to perform biological sequence analysis, and is fundamental to other applications in bioinformatics. Any multiple sequence alignment that is gradually built up by aligning pairwise sequences is essentially based on high-quality pairwise sequence alignments. By modifying common multiple alignment method based on our algorithm as shown in this paper, the accuracy was improved significantly. Moreover, we think that our technique is applicable to not only global alignment, but also some others such as, local homology

Table 5 Average Q scores on the BAliBASE 3.0 database.

Method	Reference 1 Equidistant Sequences		Reference 2 Family with "Orphans"	Reference 3 Divergent subfamilies	Reference 5 Large Insertions
	V1:0-20%ID	V2:20-40%ID			
TCoffee-MTRAP	**0.752**	**0.943**	**0.947**	**0.873**	**0.892**
TCoffee-Lu/Sze	0.695	0.937	0.939	0.851	0.879
TCoffee	0.692	0.936	0.940	0.849	0.874
MAFFT	0.722	0.901	0.945	0.864	0.900
DIALIGN*	0.566	0.860	0.883	0.766	0.861
MUSCLE	0.743	0.931	0.941	0.870	0.872
ClustalW2	0.654	0.903	0.922	0.821	0.805
Probcons	**0.811**	**0.951**	**0.957**	**0.905**	**0.909**
Probalign	0.728	0.947	0.945	0.876	0.893

The average Q scores of five reference BBS sets (described as being trimmed to homologous regions) on BAliBASE 3.0 are shown. For each reference, the better scores of the TCoffee modified by Lu/Sze algorithm (TCoffee-Lu/Sze) and the TCoffee modified by MTRAP algorithm (TCoffee-MTRAP) are in bold. The best scores of the other methods are also in bold. ID means a sequence identity of the reference alignment.

*DIALIGN reported critical errors for some testing data. Therefore, the scores of DIALIGN were calculated by the partial testing data.

search and motif-finding, which will be our future works.

Availability and requirements

Project name: MTRAP
Project home page: http://www.rs.noda.tus.ac.jp/%7Eohya-m/
Operating systems: Linux, UNIX
Programming languages: C++
License: BSD license

Authors' contributions

We three (TH, KS, MO) discussed all fundamental parts together. In details, mathematical idea mainly comes from MO and TH did mathematical algorithm. Moreover, TH and KS made computer algorithm and did computer alignment by means of this algorithm. All authors have read and approved the final manuscript.

Received: 18 August 2009 Accepted: 8 May 2010 Published: 8 May 2010

References

1. Pearson W, Lipman D: **Improved tools for biological sequence comparison.** *Proceedings of the National Academy of Sciences* 1988, **85(8)**:2444-2448.
2. Altschul S, Gish W, Miller W, Myers E, Lipman D: **Basic local alignment search tool.** *Journal of molecular biology* 1990, **215(3)**:403-410.
3. Thompson JD, Higgins DG, Gibson TJ: **CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice.** *Nucleic Acids Res* 1994, **22**:4673-4680.
4. Morgenstern B: **DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment.** *Bioinformatics* 1999, **15**:211-218.
5. Notredame C, Higgins DG, Heringa J: **T-Coffee: A novel method for fast and accurate multiple sequence alignment.** *J Mol Biol* 2000, **302**:205-217.
6. Katoh K, Misawa K, Kuma K, Miyata T: **MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform.** *Nucleic Acids Res* 2002, **30**:3059-3066.
7. Edgar RC: **MUSCLE: multiple sequence alignment with high accuracy and high throughput.** *Nucleic Acids Res* 2004, **32**:1792-1797.
8. Do C, Mahabhashyam M, Brudno M, Batzoglou S: **ProbCons: probabilistic consistency-based multiple sequence alignment.** *Genome Research* 2005, **15(2)**:330.
9. Roshan U, Livesay D: **Probalign: multiple sequence alignment using partition function posterior probabilities.** *Bioinformatics* 2006, **22(22)**:2715.
10. Feng D, Doolittle R: **Progressive sequence alignment as a prerequisite to correct phylogenetic trees.** *Journal of Molecular Evolution* 1987, **25(4)**:351-360.
11. Blackshields G, Wallace I, Larkin M, Higgins D: **Analysis and comparison of benchmarks for multiple sequence alignment.** *In Silico Biology* 2006, **6(4)**:321-339.
12. Wang K, Samudrala R: **Incorporating background frequency improves entropy-based residue conservation measures.** *BMC bioinformatics* 2006, **7**:385.
13. Gotoh O: **Consistency of optimal sequence alignments.** *Bulletin of Mathematical Biology* 1990, **52(4)**:509-525.
14. Anfinsen CB: **Principles that govern the folding of protein chains.** *Science* 1973, **181**:223-230.
15. Gonnet G, Cohen M, Benner S: **Analysis of amino acid substitution during divergent evolution: the 400 by 400 dipeptide substitution matrix.** *Biochemical and Biophysical Research Communications* 1994, **199**:489-489.
16. Crooks G, Green R, Brenner S: **Pairwise alignment incorporating dipeptide covariation.** *Bioinformatics* 2005, **21(19)**:3704.
17. Lu Y, Sze S: **Improving accuracy of multiple sequence alignment algorithms based on alignment of neighboring residues.** *Nucleic Acids Research* 2009, **37(2)**:463.
18. Thompson JD, Koehl P, Ripp R, Poch O: **BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark.** *Proteins* 2005, **61**:127-136.
19. Altschul S: **Amino acid substitution matrices from an information theoretic perspective.** *J Mol Biol* 1991, **219**:555-565.
20. Van Walle I, Lasters I, Wyns L: **SABmark - a benchmark for sequence alignment that covers the entire known fold space.** *Bioinformatics* 2005, **21(7)**:1267.
21. Murzin A, Brenner S, Hubbard T, Chothia C: **SCOP: a structural classification of proteins database for the investigation of sequences and structures.** *Journal of molecular biology* 1995, **247(4)**:536-540.
22. Lipman D, Altschul S, Kececioglu J: **A tool for multiple sequence alignment.** *Proceedings of the National Academy of Sciences* 1989, **86(12)**:4412-4415.
23. Ohya M, Uesaka Y: **Amino acid sequences and DP matching:a new method of alignment.** *Information Sciences* 1992, **63**:139-151.
24. Needleman SB, Wunsch CD: **A general method applicable to the search for similarities in the amino acid sequence of two proteins.** *Journal of Molecular Biology* 1970, **48**:443-453.
25. Rice P, Longden I, Bleasby A: **EMBOSS: the European Molecular Biology Open Software Suite.** *Trends Genet* 2000, **16**:276-277.
26. Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ, Higgins DG: **Clustal W and Clustal X version 2.0.** *Bioinformatics* 2007, **23**:2947-2948.
27. Mizuguchi K, Deane CM, Blundell TL, Overington JP: **HOMSTRAD: a database of protein structure alignments for homologous families.** *Protein Sci* 1998, **7**:2469-2471.
28. Stebbings L, Mizuguchi K: **HOMSTRAD: recent developments of the homologous protein structure alignment database.** *Nucleic acids research* 2004, , **32 Database:** D203.
29. Thompson J, Plewniak F, Poch O: **A comprehensive comparison of multiple sequence alignment programs.** *Nucleic Acids Res* 1999, **27(13)**:2682-2690.
30. Sauder J, Arthur J, Dunbrack R Jr: **Large-scale comparison of protein sequence alignment algorithms with structure alignments.** *Proteins Structure Function and Genetics* 2000, **40**:6-22.

doi:10.1186/1471-2105-11-235
Cite this article as: Hara *et al.*: MTRAP: Pairwise sequence alignment algorithm by a new measure based on transition probability between two consecutive pairs of residues. *BMC Bioinformatics* 2010 **11**:235.